Amit Agarwal (Cal. #294269)
P.O. Box 10354
Tampa, FL 33679
813-955-3949
ama7386@gmail.com
Counsel for Defendant Fleksy, Inc.

UNITED STATES DISTRICT COURT

NORTHERN DISTRICT OF CALIFORNIA, OAKLAND DIVISION

WORDLOGIC CORPORATION, et al.,

Plaintiff,

v.

FLEKSY, INC.,

Defendant.

No. 4:17-CV-7169-JSW

DEFENDANT FLEKSY'S NOTICE OF MOTION AND MOTION FOR SANCTIONS UNDER RULE 11

Judge Jeffrey White

June 1, 2018 at 9:00 a.m.

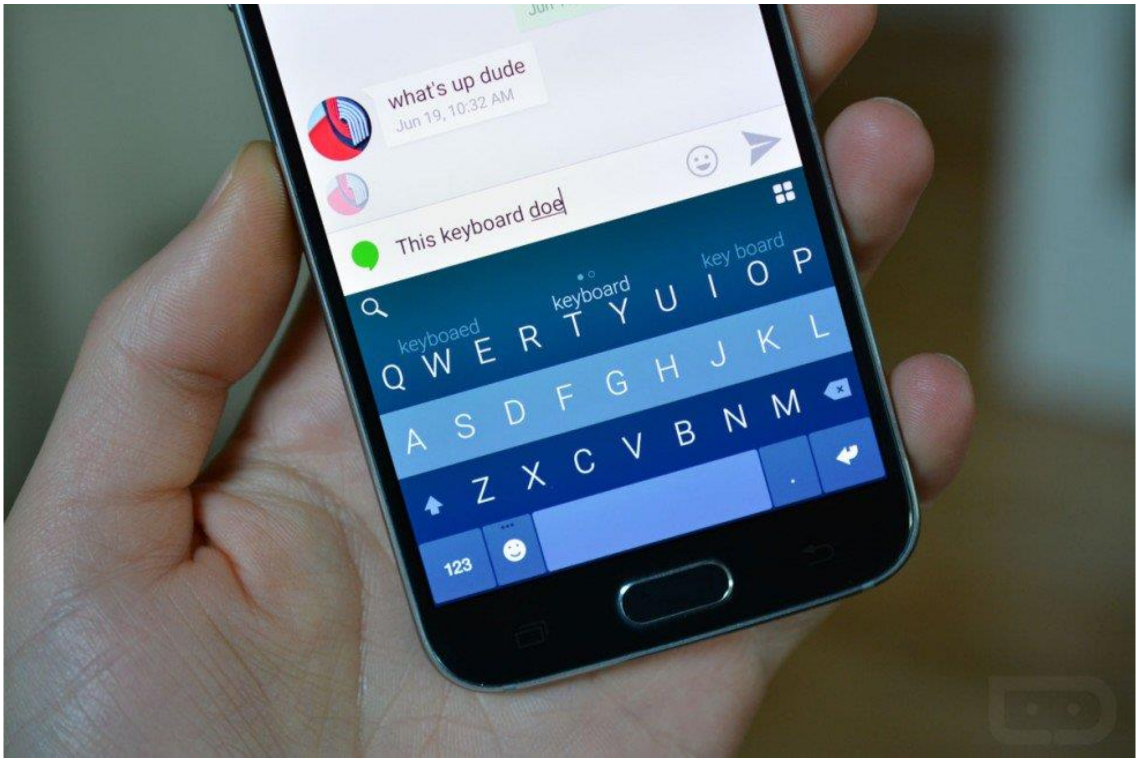
Courtroom 5, 2nd Floor

Notice of Motion, Motion, and Relief Requested

On May 18, 2018 at 9 a.m. or as soon thereafter as this motion can be heard by J. Jeffrey White in Courtroom 5, U.S. District Court for NDCA, 1301 Clay St., Oakland CA 94612, Defendant Fleksy, Inc. ("Fleksy") will and hereby moves the Court, under FRCP 11, to direct Plaintiffs and their counsel to pay a monetary sanction equal to the reasonable fees that Fleksy has spent defending this meritless lawsuit which should never have been filed in the first place.

1. Introduction

The accused product is a free smartphone app called "Fleksy." It is superior to whatever default keyboard apps come pre-installed with smartphones.



It would've taken zero dollars and thirty seconds to verify that this app does not practice, literally or under DoE, entire claim elements recited in the only two patent claims mentioned in the Complaint—claim 19 of Pat. 7,681,124 ("'124") and claim 1 of Pat. 8,552,984 ("'984). Yet, that didn't stop Plaintiffs and their counsel from paying a $400 filing fee and imposing $30,000+ in legal defense costs (to date, and mounting). This Court can identify the meritless nature of this

Complaint by reviewing the opposition brief in detail which will contain nothing that could assure this Court that discovery was *ever* likely to reveal evidence of patent infringement.

2. Plaintinffs (and their counsel's) assertion of '124 Patent Claim 19 is meritless.

Claim 19 recites:

> A computer-readable medium comprising codes for directing a processing unit to process text entered into a personal computing device, by:
>
> > (a) receiving and displaying a partial text entry, comprising receiving at least part of the partial text entry via a keyboard, the partial text entry comprising at least a first character;
> >
> > (b) in response to receipt of the first character of the partial text entry, obtaining a plurality of completion candidates from among a group of completion candidates, wherein each of the plurality of completion candidates includes a portion matching the partial text entry;
> >
> > (c) displaying the plurality of completion candidates in a search list within a graphical user interface;
> >
> > (d) detecting user input corresponding to selection of a particular completion candidate from among the plurality of completion candidates displayed in the search list;
> >
> > (e) modifying the display of the partial text entry to correspond to the particular completion candidate selected from among the plurality of completion candidates at least while the particular completion candidate remains selected;
> >
> > (f) detecting modification of the partial text entry by the user via the keyboard;
> >
> > (g) obtaining and displaying in the search list a modified plurality of completion candidates from among the group of completion candidates, if the partial text entry is modified via the keyboard, wherein each of the modified plurality of completion candidates includes a portion matching the partial text entry; and
> >
> > (h) obtaining and displaying in the search list a further modified plurality of completion candidates from among the group of completion candidates, if a completion candidate is accepted via the search list from the modified plurality of completion candidates, wherein each of the further modified plurality of completion candidates includes a portion matching the accepted completion candidate.

The color codes help to break down this claim into its three parts.

(1) **Basic text entry**: Receiving a partial text entry, displaying a list of completion

candidates, then receiving the user's selection of a completion candidate (steps a—e);

(2) **Modifications**: Detecting the modification of a partial text entry and obtaining and displaying a modified completion candidate list (steps f—g); and

(3) **Multi-level search lists**: Obtaining and displaying a further modified completion candidate list *if* a completion candidate is *accepted* (step h), something the specification calls "multi-level search lists." On this third point corresponding to claim step (h), the specification elaborates:

> In one embodiment, when the search list is active the user may use one of the completion candidates in the search list to initiate a further automated search to obtain a more refined list of completion candidates. In this embodiment, multi-level search lists and searching are available to help accelerate completion of a partial text entry. As a result, the user can automatically initiate an iterative search wherein a completion candidate listed in the search list is used as the new partial text entry to dynamically obtain a new list of completion candidates, which is then displayed in the search list. The automated ability to use the search list to obtain a refined list of completion candidates allows the user to quickly make good use of search results that are only partially successful.
>
> '124 at 2:57-3:2.
>
> In the third case, multi-level search lists and searching are available to help accelerate completion of a partial text entry. In this latter case, the user can automatically initiate an iterative search wherein a completion candidate listed in the interactive search list is used as the new partial text entry to dynamically obtain a new list of completion candidates, which is then displayed in an updated interactive search list. The automated ability to use the interactive search list to dynamically obtain a refined list of completion candidates allows the user to quickly make good use of search results that are only partially successful. When the interactive search list is updated with a new list of completion candidates, the user can then choose one of the completion candidates in the new list, or the user can repeat the iterative search process once again by choosing one of the completion candidates in the new list and activating a further iterative search.
>
> '124 at 6:30-45.

Wordlogic's counsel certified under Rule 11 (by filing this Complaint) that he had evidentiary support or, if specifically so identified, will likely have evidentiary support after a reasonable opportunity for further investigation or discovery, for the factual contention that Fleksy's app obtains and displays a list of further modified completion candidates *following* the user's acceptance of a completion candidate that includes a portion of the accepted completion candidate.

Fleksy invites the Court's attention to the brief filed in opposition to this motion. It is a guarantee that there will be nothing in the realm of *evidence* that Fleksy does this. But, more importantly, counsel for Plaintiffs will be unable to cite anything *non-evidentiary* to convince this Court that he had a "good faith" basis to believe that the app he accused of infringing this patent claim actually satisfies the recited element. A thirty-second review of the app would have confirmed that multi-level search or anything close to it simply <u>does not occur</u> and has never occurred on Fleksy's app. There isn't an iota of evidence in the public domain that could plausibly support a reasonable belief that discovery would uncover such evidence. Any bare bones pre-filing investigation would have revealed to counsel that on Fleksy's app, when a user accepts a completion candidate, the app inserts the word, then inserts a space bar after the word, and then awaits the user's entry of a brand new word. It does <u>not</u> obtain and display a list of further completion candidates. That is, the Fleksy app does not process a user selection of a completion candidate as a request by the user to continue searching for *further* completion candidates, *i.e.*, multi-level search of the sort explained in the '152 patent specification and claimed in claim element (h) of Claim 19.

To be sure, Fleksy has not shared a line of source code with Wordlogic or its counsel. But Rule 11 applies at the Complaint stage because a federal complaint contains a set of pleadings that are governed by Rule 11. That is, even with zero discovery, counsel for Wordlogic <u>must</u> have either evidentiary support or a reasonable belief that he will likely have evidentiary support after a reasonable opportunity for further investigation or discovery. Fleksy's motion challenges Wordlogic to submit <u>anything</u> in its opposition brief which can offer assurance to this tribunal that the results of Wordlogic's counsel's pre-filing investigation render this Complaint compliant under Rule 11. It cannot be the case that the accused product feature's mere title, *word prediction*, is sufficient for a reasonable person to believe that it probably performs the specific multi-level word prediction function recited in claim element (h) of claim 19. The sheer specificity of this sub-function counters the proposition that the patentee has an entitlement to discovery in federal litigation *just because* someone in the universe has a keyboard app that performs word completion. It would be frivolous to contend that the claims of these patents render all word prediction software suspect under direct infringement.

Step (h) recites "**obtaining and displaying in the search list a further modified plurality of completion candidates** from among the group of completion candidates, **if a completion candidate is accepted** via the search list from the modified plurality of completion candidates, wherein each of the further modified plurality of completion candidates includes a portion matching the accepted completion candidate." That is, if the event in red occurs, the invention responds with the action in blue. This is not a signature feature in word-prediction technology. This is a highly specific claim element which an attorney cannot *believe*—in good faith—probably exists in all word prediction apps such that with zero evidentiary basis he can assert that discovery will reveal something useful.

A demand for source code is likewise unavailing. The sheer nature of this product is that one can confirm or disconfirm the satisfaction of claim element (h) by merely *using* the product. Source code is irrelevant for this case. But even if it were relevant—again—patentees have no entitlement to discovery into source code for *an entire industry* just because they hold a patent for one specific invention within that industry and survived a Rule 12 motion.

Consider a very specific encryption algorithm for which the PTO grants a patent award to an entity. Can that entity go around the country suing every single encryption company with *zero* basis to believe that those companies perform his particular encryption algorithm? The *only* plausible rationale for even entertaining the notion of giving discovery to this hypothetical patentee is that in encryption, it is perhaps impractical to observe manifestations of the end product and ascertain infringement versus non-infringement without discovery into proprietary code. But that barely coherent rationale is entirely missing here, where the app was costless and instantly downloadable. The very nature of these claims and the nature of the product is that it was designed for laypersons. The claims do not patent some inner functionality of the hardware.

They speak to the interface with the user.

For these reasons, Plaintiffs' counsel's assertion of direct infringement of Claim 19 of the '124 patent is a violation of FRCP 11.

3. Plaintiff's (and its counsel's) assertion of Count 2 of the Complaint violates Rule 11.

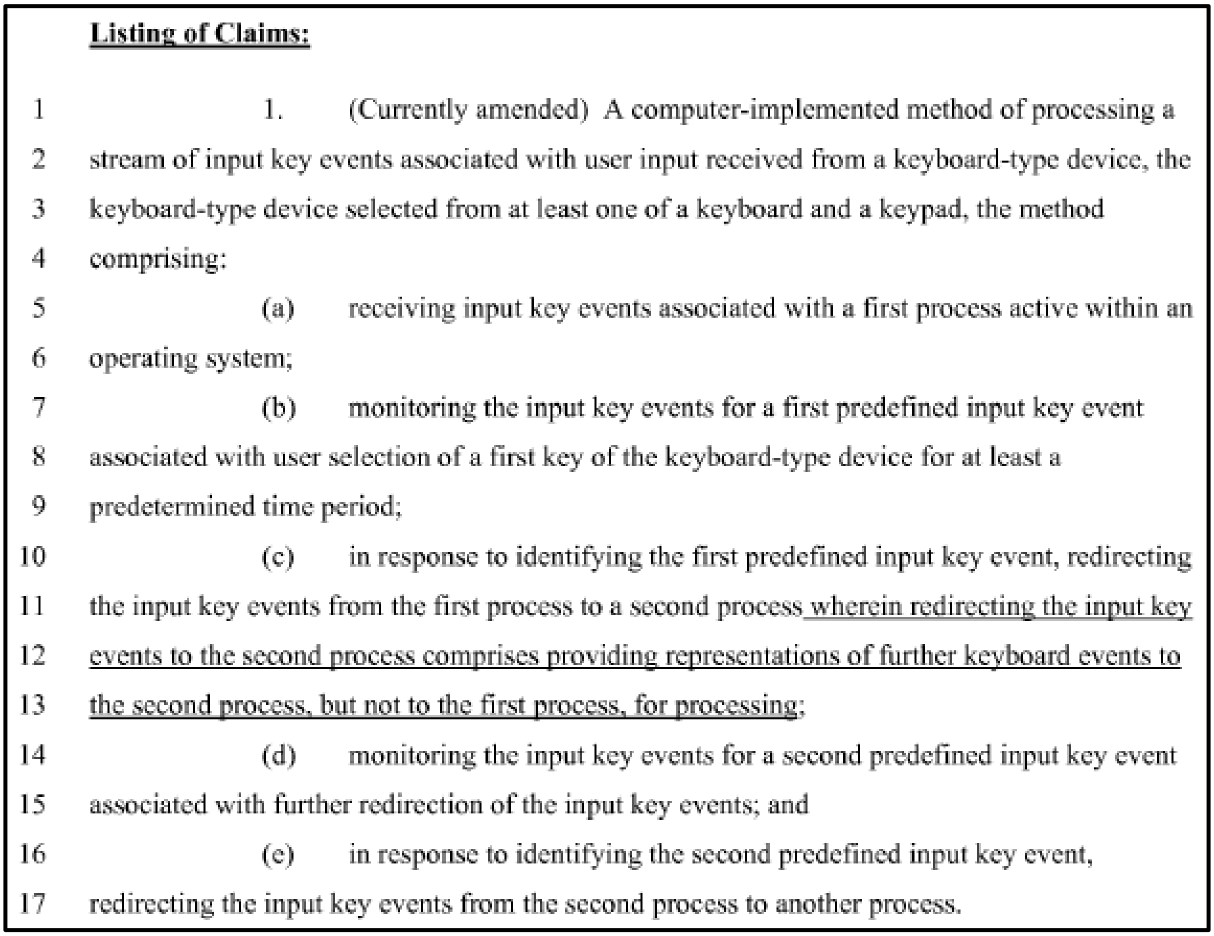
The second count set forth in the Complaint is a count of direct infringement of at least claim 1 of the '984 patent. Complaint at ¶¶ 23-28. Wordlogic alleges that the "Fleksy predictive keyboard redirects key input from an application in which text is being entered to the keyboard application itself when a particular key is held for a predetermined amount of time, and then redirects key input back to the original application upon selection of a second key." *Id.* at ¶ 25. Wordlogic's claims and legal contentions relating to direct infringement of claim 1 (or any other claim, for that matter) of the '984 patent are **not** warranted by existing law or by a nonfrivolous argument for extending, modifying, or reversing existing law or for establishing new law. This is a Rule 11 violation.

Claim 1 of the '984 patent, the only claim mentioned from that patent in the Complaint, recites:

> 1. A computer-implemented method of processing a stream of input key events associated with user input received from a keyboard-type device, the keyboard-type device selected from at least one of a keyboard and a keypad, the method comprising:
>
> > (a) receiving input key events associated with a first process active within an operating system;
> >
> > (b) monitoring the input key events for a first predefined input key event associated with user selection of a first key of the keyboard-type device for at least a predetermined time period;
> >
> > (c) in response to identifying the first predefined input key event, ***redirecting the input key events from the first process to a second process wherein redirecting the input key events to the second process comprises providing representations of further keyboard events to the second process, but not to the first process***, for processing;
> >
> > (d) monitoring the input key events for a second predefined input key event associated with further redirection of the input key events; and

> (e) in response to identifying the second predefined input key event, redirecting the input key events from the second process to another process.

During an office action mailed on Dec. 31, 2012, the examiner rejected the applicant's attempt to claim step (c) above *without* the boldfaced amendment citing Dostie et al. (WO 02/33527) under 35 U.S.C. § 102(b). File History at 104. In response to this office action, on April 30, 2013, the applicant surrendered claim scope by narrowing his claims to include the added limitation underlined below. Ex. A at 69.



Listing of Claims:

1. (Currently amended) A computer-implemented method of processing a stream of input key events associated with user input received from a keyboard-type device, the keyboard-type device selected from at least one of a keyboard and a keypad, the method comprising:

(a) receiving input key events associated with a first process active within an operating system;

(b) monitoring the input key events for a first predefined input key event associated with user selection of a first key of the keyboard-type device for at least a predetermined time period;

(c) in response to identifying the first predefined input key event, redirecting the input key events from the first process to a second process wherein redirecting the input key events to the second process comprises providing representations of further keyboard events to the second process, but not to the first process, for processing;

(d) monitoring the input key events for a second predefined input key event associated with further redirection of the input key events; and

(e) in response to identifying the second predefined input key event, redirecting the input key events from the second process to another process.

Compare claim step (c) with Wordlogic's pleadings in the Complaint.

Claim element from cl. 1 of the '984 patent	**Complaint allegation**
"redirecting the input key events from the first process to a **second process** wherein redirecting the input key events to the second process comprises providing representations of further keyboard events	The Fleksy predictive keyboard redirects key input from an application in which text is being entered to the **keyboard application itself** when a particular key is held for a predetermined amount of time, and then redirects key input back to the

to the second process, **but not to the first process**"	original application upon selection of a second key." Complaint at ¶ 25.

Again, how can we claim a Rule 11 violation without providing a single line of source code?

Not every app which redirects input key events from a first process to a second process necessarily satisfies the purple claim term. Specifically, apps in which the redirection of input key events comprises providing representations of further keyboard events to the second process and also to the first process remove those apps from the plain and ordinary meaning of the phrase "but not to the first process." A few minutes of due diligence would have sufficed to reveal that in the Fleksy predictive keyboard, representations of keyboard events are always provided to the first process, *i.e.*, the application in which text is being entered. Notably, the "second process," as the Complaint itself acknowledges is the *keyboard app itself*. It takes faint introspection to conclude that input into a keyboard is—based on the definition of a keyboard—being provided to the app that the keyboard is being used to enter input into. That is, the basic and only function of a keyboard (either a hardware keyboard or a software keyboard) is to provide representations of keyboard events to the underlying application.

One might wonder why the claim is drafted in this manner. Why does the patentee *require* the negative claim limitation "but not to the first process"? Doesn't this rob the patent claim of all practical utility? Are there examples of "second processes" where it would make more sense to not send the input to the "first process"? Look no further than the specification and prosecution history to provide a crystal clear example *and* explanation of the advantage of the claimed invention.

Specification: "Enabling the redirection of keyboard input to be independent of which process has keyboard focus at the operating system level provides a significant advantage in that it allows the input management system 20 to support redirection of the keyboard input to one or more processes independent of limitations imposed by the operating system **22** on keyboard focus at the operating system level." '984 at 7:14-20. "[T]he second process 34 serves as a function expander, providing additional functionality relevant to the operation of the first process 30.” *Id.* at 16:12-15

File history:

> To illustrate one advantage, as disclosed in Figure 11 and para. [0129], a text editor application (i.e., a first process) lacking a thesaurus can be enhanced by enabling a user typing on a keyboard, to trigger a mid-stream redirection of the keyboard input stream to a thesaurus application (i.e., a second process) such that the text editor does not receive the redirected keyboard input stream. For example, a user may be half-way through typing a description of a landscape. The user wants to describe "greenery" but considers the word "greenery" to be too prosaic to use. The user can seamlessly invoke a thesaurus, even though the text editor lacks a thesaurus. While typing in the **text editor application** (**first process**), the user holds down the space bar for two seconds (i.e., at least a predetermined time limit) to trigger redirection of the input stream to the thesaurus application (**second process**). **The user types "greenery" to search the thesaurus, which returns a number of synonyms, including the word "verdure", which the user likes**. The user selects "verdure" and cancels the redirection mode. "Verdure" is sent to the text editor (first process). Any further text typed by the user, now goes to the text editor.
>
> **Advantageously, the word "greenery" is not entered into the text editor, even though it was typed while input key events were redirected to the second process.** The user did not want to type the word "greenery" in the text editor, but did want to send it to a function expanding (second) process. In another scenario, the user may wish to switch from typing numbers into a text editor (first process), to typing calculations into a calculator application (i.e., the second process), but to have only the results (not the details) of the calculation sent to the text editor.
>
> **Contra the Office action, providing keyboard events to both the first and second process, would not perform equally as well as providing keyboard events to the second process but not to the first process. If the word "greenery" was sent to both the text editor and the thesaurus, the user would have to delete the word "greenery" in the text editor. Similarly, the user may want only the result of a calculation entered in the text editor (e.g., "42") and not the details ("294/7").**

Ex. A at 134 (file history).

The most critical phrase in the above disclosure is the phrase “the user would have to delete the word ‘greenery’ in the text editor.” It captures the *advantage* that the applicant cited for the “not to the first process” claim element. The applicant explained that keeping input contained within the second process *to the exclusion of the first process* is advantageous because it allows the user to use the second process as a silo of sorts, adding its own independent functionality, *e.g.*, dictionary, thesaurus, calculator. The intermediate input *into that second process* has no business going to the underlying first process. Sending such intermediate input, *e.g.*, operands where a calculator is the second process or word queries where a dictionary/thesaurus is the second process. If that intermediate input is sent to the first process, as the applicant explains, the user would have to manually delete it every single time.

This is simply not the case with Fleksy's expanded keyboard serving as the second process. Every single input to that process, by definition, must go to the underlying first process. And lest the Court thinks the doctrine of equivalents is a viable option, note that Wordlogic narrowed an originally broader claim in response to a rejection to add this very claim element—"not to the first process." Wordlogic may not argue that the surrendered territory comprised unforeseen subject matter that should be deemed equivalent to the literal claims of the issued patent. Again, zero dollars and thirty seconds is *all* it would have taken for Plaintiffs' counsel to avoid fourteen months and $31,000 of litigation over these frivolous assertions.

4. Lawyers asserting software patent claims are not exempt from their Rule 11 obligations merely because they do not have access to a defendant's source code.

Wordlogic's counsel *may* argue that because discovery has not begun in this matter, that it does not yet have access to Fleksy's source code. On this basis, counsel might argue that it is premature to pass judgment on the frivolity of its patent assertions.

This argument, if made, would entirely miss the point of Rule 11. Rule 11 is *only* concerned with whether—at the time a pleading such as a complaint was presented to the Court—that the claims and legal contentions *then made* are nonfrivolous. Likewise, as to factual contentions, Rule 11 is *only* concerned with whether the lawyer certifying the counts of patent infringement has evidentiary support or will likely have evidentiary support after a reasonable opportunity for further investigation or discovery.

Here, the accused product is a mobile app that was freely and easily available for download and inspection at the time Wordlogic's counsel filed this patent lawsuit. For zero dollars and ten minutes of his time, counsel could have confronted the factual realities about Fleksy's app that Fleksy has detailed in this brief. These factual realities collide head first with entire claim elements recited in the two patents-in-suit. Source code, in this context, is entirely irrelevant when counsel for Wordlogic was perfectly capable of inspecting the accused product and observing on the user interface entirely reflective of such code that:

1. The Fleksy app does not practice "obtaining and displaying in the search list a further modified plurality of completion candidates from among the group of completion candidates, if a completion candidate is accepted via the search list from the modified plurality of completion candidates, wherein each of the further modified plurality of completion candidates includes a portion matching the accepted completion candidate" and therefore cannot infringe claim 19 of the '124 patent.
2. The Fleksy app *always* provides representations of input key events to the first process and therefore cannot infringe claim 1 of the '984 patent.

Neither of these observations require examination of a single line of source code.

5. The relationship of this Rule 11 to a previously filed Rule 12 motion.

With respect to the '124 patent, weeks into this case which has now gone on for over a year, Fleksy filed a Rule 12 motion attempting to weed this case out at inception. The transferor court ruled that "The '124 patent is directed to predictive text entry. It is reasonable to infer that a completion candidate, which attempts to complete partially entered text, must include a portion matching the text a user has entered. When a user modifies the entered text, the logical inference is that a list of modified completion candidates, which likewise seek to complete partially entered text, must include a portion matching the modified text. Accordingly, plaintiffs have satisfied their pleading burden with respect to the '124 patent." ECF No. 30 at 4. This *does not account* for the multi-level search recited in claim element 19(h) of the '124 patent as explained in this brief. Besides, the standard for Rule 12 has an incomplete overlap with Rule 11.

As this Court is well aware, "[a] patent suit can be an expensive proposition. Defending against baseless claims of infringement subjects the alleged infringer to undue costs—precisely the scenario Rule 11 contemplates." *View Eng'g, Inc. v. Robotic Vision Sys., Inc.*, 208 F.3d 981, 986 (Fed. Cir. 2000). Patent cases are burdensome on the parties and the Court. Rule 11 is designed to weed out unmeritorious cases before those burdens are imposed—by causing parties and their counsel to "stop-and-think" before making legal or factual contentions. Fed. R. Civ. P. 11 advisory comm. n. 1993 Am.

6. Conclusion

This lawsuit is frivolous. The opposition brief *itself* will confirm that for this Court because the evidence, logic, reason, and rhetoric will provide zero support for the hypothesis that one could harbor a good faith expectation that discovery would reveal evidentiary support for the two particular claim elements highlighted in this brief. This is the stuff of Rule 11. This Court should therefore award Rule 11 sanctions equal to the reasonable attorney's fees spent defending this action.

March 23, 2018

/s/Amit Agarwal
Amit Agarwal
California Bar #294269
P.O. Box 10354
Tampa, FL 33679
813-955-3949
ama7386@gmail.com
Counsel for Defendant Fleksy, Inc.

CERTIFICATE OF SERVICE

I hereby certify that on March 23, 2018, the within document was served via EM/ECF to all participants including Mr. Anthony Dowell who represents Plaintiffs.

/s/ Amit Agarwal

Amit Agarwal